**Claims:**

1.  A method for determining erasures in an FEC (Forward Error Correction) decoding process decoding data encoded with concatenated codes, the method comprising the steps of:

    generating first output data by decoding first input data;

    generating (51) second output data by decoding second input data, the second input data including at least a part of the first output data;

    comparing (52) the first and the second output data;

    updating (53) a comparison result accumulation parameter based on the comparison result; and

    determining (54), based on the updated comparison result accumulation parameter, whether an erasure is to be set.

2.  The method according to claim 1, wherein the first and second output data are generated in different or the same decoder stages.

3.  The method according to claim 1 or 2, wherein the erasure is set in the input data for a subsequent step of generating decoded output data.

4.  The method according to one of claims 1 to 3, wherein the FEC decoding process uses a sequence of decoder stages, each subsequent decoder stage receives at least a part of the output data of a previous decoder stage.

5.  The method according to one of claims 1 to 4, wherein the FEC decoding process is performed in an iterative manner.

6. The method according to one of claims 1 to 5, wherein the step of comparing is performed by evaluating whether respective symbols in the first and the second output are equal.

5   7. The method according to one of claims 1 to 6, wherein the step of comparing is performed at least for two different pairs of output data, and wherein the comparison result accumulation parameter stores the comparison results for corresponding previous comparisons.

10   8. The method according to one of claims 1 to 7, wherein the step of comparing is performed between first and second decision data each included in the respective output data.

9. The method according to one of claims 1 to 8, wherein the first and second
15   output data each comprise extrinsic data, the extrinsic data in the first output data being the part used as the input for generating the second output data.

10. The method according to one of claims 1 to 9, wherein the erasure is set in a part of the second output data, which is used as an input for a subsequent
20   step of generating decoded output data.

11. The method according to one of claims 1 to 10, wherein at least the step of comparing is performed per bit or symbol of the output data.

25   12. The method according to one of claims 1 to 11, further comprising: generating third output data by decoding third input data, the third input data including at least a part of the second output data, comparing the third and the second output data, further updating the comparison result accumulation parameter based on the comparison result between the third and the second output data,
30   and determining, based on the further updated comparison result accumulation parameter, whether an erasure is to be set.

13. The method according to one of claims 1 to 12, wherein the step of updating the comparison result accumulation parameter is performed by modifying a

counter when the comparison result indicates that the compared output data are unequal, and wherein the step of determining includes comparing the comparison result accumulation value with a predefined comparison result threshold that depends on the number of respective output data comparison steps performed.

14. The method according to one of claims 5 to 13, wherein the steps of comparing and determining are performed after each decoding stage, or after each decoding stage subsequent to a minimum number of decoding stages, or after a predetermined number of decoding stages.

15. The method according to one of claims 5 to 14, further comprising performing an error statistic after the final decoding stage, determining a specific threshold value based on the error statistic result, and in the step of determining, which is performed after the final decoding stage, using the specific threshold as a decision criterion.

16. The method according to one of claims 1 to 15, wherein the decoding process decodes data encoded with a Turbo Code as an inner code and a Reed Solomon Code as an outer code.

17. The method according to one of claims 1 to 16, wherein the erasure is a log-likelihood ratio having the value zero.

18. The method according to one of claims 5 to 17, wherein the step of determining defines erasures for systematic data or parity data input to a subsequent decoding stage.

19. The method according to one of claims 1 to 18, wherein the step of determining further includes a step of checking whether a soft metric of the output data is smaller than a soft metric threshold.

20. The method according to claim 19, wherein the soft metric threshold is determined based on a decision error probability.

21. An FEC (Forward Error Correction) decoder device for decoding data FEC encoded with concatenated codes, the FEC decoder comprising:

5       decoding means arranged for generating first output data by decoding first input data and generating second output data by decoding second input data, the second input data including at least a part of the first output data; and

an erasure decision unit arranged for evaluating whether the first and the

10      second output data are equal, for updating a result accumulation parameter based on the evaluation result and a previous value of the result accumulation parameter, and for determining, based on the comparison result accumulation parameter, whether an erasure is to be set.

15  22. The FEC decoder device according to claim 21 adapted to operate in accordance with a method of claims 1 to 20.

23. The FEC decoder device according to claim 21 or 22, wherein the decoding means includes a multistage turbo decoder.

20

24. A receiver for use in a communication system, the receiver comprising the FEC decoder device according to one of claims 21 to 23.

25. A communication device comprising the FEC decoder device according to

25      one of claims 21 to 23.